

УДК 004.438

Студ. Н. А. Лаптев, М. В. Дубовик

Науч. рук. ассист. И. А. Миронов

(кафедра информатики и веб-дизайна, БГТУ)

## НОВЫЙ ЯЗЫК ПРОГРАММИРОВАНИЯ JAI

В текущий момент программирование игр является одним из направлений программирования в целом. Для такого типа программирования очень важна скорость обработки кода, т.к. графика в играх очень динамична и пользователю будет неприятно работать с временными задержками. Для выигрыша в скорости удобно использовать низкоуровневые языки, такие как язык ассемблера, но они очень объёмные по коду, поэтому для разработки обычно выбор падает на языки среднего уровня, такие как C/C++.

В 2014 году разработчик игр Джонатан Блоу предложил концепцию нового языка программирования Jai, ориентированного преимущественно на разработку игр. Компилятор под данный язык на текущий момент разрабатывается Блоу. Автор анонсировал, что новый язык будет выигрывать в скорости обработки кода у языка C. Данный факт заставляет рассмотреть данный язык более подробно.

**Jai** – высокоуровневый, процедурный, строго типизированный, C-подобный язык программирования. Описание языка предполагает его как современную замену языку C.

Среди особенностей языка можно отметить:

- Произвольное выполнение кода во время компиляции. Любая функция программы может быть запущена во время компиляции с помощью *#run*;
- рефакторинг кода с облегченным синтаксисом. Синтаксис языка облегчает повторное использование кода, упрощая его перемещение из *локального блока* → *локальной функции* → *глобальной функции*;
- интегрированный процесс сборки. Процесс сборки и параметры задаются самим исходным кодом для обеспечения согласованности;
- автоматическое преобразование между структурой массивов и массивом структур, избегает классы и наследование;
- рефлексия (процесс, во время которого программа может отслеживать и модифицировать собственную структуру и поведение во время выполнения) и информация о типе времени выполнения. Информация о статическом типе для каждой структуры доступна во время выполнения;

- новый подход к полиморфным процедурам. Полиморфизм на функциональном уровне, с программным управлением специальными процедурами;

- низкоуровневые инструменты управления памятью. Более эффективный контроль над тем, как библиотеки выделяют память, автоматическое управление собственностью, сборка мусора отсутствует;

- явный контроль над оптимизацией и производительностью. Явный контроль над такими вещами, как встраиваемые функции (inline-функции), «проверка границ» и инициализация.

Основная идея Блоу состоит в том, что видеоигры – это огромные блоки памяти, которые перемещаются последовательно с жёсткого диска в оперативную память, с оперативной памяти загружаются в видеокарту или кэш процессора для обработки, а далее возвращаются в оперативную память. Большое время ожидания выполнения этих процессов не нравится пользователям данных игр. Автор сказал, что использование сборщиков мусора, шаблонов и классов помогают быстрее писать код программистам, но это не делает их код более быстрым.

Особо привлекательной выглядит попытка оптимизации памяти при работе с массивами структур. Её суть состоит в том, что массивы структур при явном указании соответствующего ключевого слова SOA преобразуются к структуре массивов, что позволяет сократить размеры неиспользованной памяти.

Важная цель в проектировании Jaі – уменьшение того, что Блоу называет «трением» в программировании. «Трение» случается, когда синтаксис языка мешает работе программиста. Создатель языка утверждает, что увеличение производительности и уменьшение «трения» при отсутствии механизмов защиты памяти более рационально, чем восполнение времени, потраченного на отслеживание ошибок, особенно если учесть, что хорошие программисты допускают относительно небольшое количество ошибок. Поэтому в Jaі низкая терпимость к «трению», особенно когда «трение» не нужно.

Джонатан Блоу создал концепцию такого языка, который будет ориентирован на программирование игр, но, как отметил сам автор, язык может быть использован и в других направлениях программирования. Даже, если язык после создания компилятора не получит большого распространения, то идеи Джонатана Блоу могут

послужить отправной точкой для дальнейшего развития других языков программирования.

УДК 025.5:004(072)

Студ. Е.А. Богатырев

Науч. рук. доц. Романенко Д.М., ассист. Т.П. Брусенцова  
(кафедра информатики и веб-дизайна, БГТУ)

## **ОСОБЕННОСТИ РАЗРАБОТКИ КЛИЕНТСКОЙ ЧАСТИ ИНФОРМАЦИОННО-ПОИСКОВОГО СЕРВИСА В СФЕРЕ УСЛУГ**

Информационно-поисковые системы являются неотъемлемой частью жизни современного человека. С появлением Всемирной паутины и сети Интернет люди получили доступ к огромному объему информации, расположенной по всему миру. Своевременно появились специализированные информационно-поисковые системы, которые собирали, хранили, обрабатывали и осуществляли поиск по информации определенного характера: информация об отелях и свободных номерах, сеансах в кинотеатрах, постановках в театрах, концертах. Так начали появляться информационно-поисковые системы в сфере услуг.

Информационно-поисковая система – это компьютерная система, предназначенная для поиска информации. Сейчас наиболее распространенным видом таких систем является веб-приложение, построенное по архитектуре «клиент-сервер». Это обусловлено тем, что позволяет обеспечить одновременную работу большого числа пользователей. Клиентское приложение предоставляет пользовательский интерфейс, взаимодействуя с которым пользователь формирует запрос на получение нужной ему информации и получает результат.

В данный момент широкое распространение получили следующие виды приложений:

1. WEB-приложение;
2. Мобильное приложение;
3. Desktop приложение.

WEB-приложение – ПО, исполняемое в интернет браузере, использующее протоколы HTTP/HTTPS для обмена данными. Достоинствами такого приложения является независимость от операционной системы, платформы, удобство распространения приложения, обновлений. К недостаткам можно отнести потребление